**Processing**

Since we were only two students to do this project, we could not get any image processing done in time. If we had to implement processing, we would stop grabbing frames, read a chunk from memory, modify it and write it back into memory. We would have to be careful not to grab a whole frame since we don’t want to create a useless block of memory.

**Constraint file**

The full constraint file used to compile our design can be found under the name "Real constraint file.txt" in the "code.zip" archive attached to this report. Two clocks were created: 27 MHz, 50 MHz and the 100 MHz clock was generated by a PLL. False paths were then set from the 27 MHz clock to the 100 MHz clock and vice versa.

**Compilation reports**

The screenshots of the compilation reports can be found in figures 11 through 19 of the Appendix. Our design used 20% of the logic elements available as well as 56% of the memory bits (see appendix, figure LINK FLOW SUMMARY). According to the “slow model”, we could have used a maximum frequency of 103.32 MHz for the 100 MHz clock and 148.63 MHz for the 27 MHz clock (see appendix, figure LINK SLOW MODEL MAXIMUM FREQUENCY). The hold time of the 27 MHz and 100 MHz clock both have a slack of 0.391 (see appendix, figure LINK TO HOLD SUMMARY) while the 27 MHz has a slack of 30.272 in its setup time and the 100 MHz has a slack of 0.321 (see appendix, figure LINK TO SETUP SUMMARY). Note that we did not get any illegal clocks (see appendix, figure 19 LINK TO UNCONSTRAINED PATHS).

**Sources of problems**

In the demonstration, we were able to display a video from a source, though part of the screen was blurry. Afterwards, we noticed that the grab interface is not writing enough, causing the DMA to read invalid data on odd fields only (see appendix, figure LINK + ADD FIGURE Odd field doesn’t seem to work).